Fermilab

THE FERMILAB TIMELINE GENERATION SYSTEM[*]

R. P. Johnson, W. R. Knopf, and A. D. Thomas

June 1985

THE FERMILAB TIMELINE GENERATION SYSTEM

R. P. Johnson, W. R. Knopf, and A. D. Thomas
Fermilab[*], P.O.Box 500, Batavia, Il. 60510

Abstract

In this paper the technique used to control the relative timing and synchronization of the major accelerator systems at Fermilab is described.

Introduction

The various operating modes of the injector accelerators include fixed target and colliding beam operation in conjunction with simultaneous machine studies. For example, in a 60 second interval the conventional Main Ring may be called upon to: a) load the Tevatron with 12 high intensity Booster batches each containing 82 rf bunches at 150 GeV, b) transfer a Booster batch at 8 GeV with 8 rf bunches to the Debuncher or Accumulator, c) accelerate high intensity beam several times to 120 Gev for antiproton production, and d) accelerate beam to 150 GeV for Main Ring studies. In the case of colliding beam operation, the different tasks can be even more varied. All this requires a simple, flexible means of coordination.

Alternatives

One approach to solving this problem, employed at the CERN PS, would be to have front-end computers keep track of all the parameters associated with a certain machine cycle as a separate data base. As a particular cycle begins, all the front-end computers load the hardware from that data base. Likewise, a user wishing to change some parameter from a control console simply chooses that particular type of machine cycle as if it were a separate machine. Thus one may tune any machine mode independently of any others.

We have decided to approach the problem in a different manner for two reasons. The first is that it would have demanded an unacceptable expansion of the control system software during the very time the control system and the Tevatron were being commissioned. The second comes from investigation of those parameters which actually need to change for different types of cycles in the Main Ring and Booster. In fact, we believe the number of parameters and function tables which have to be swapped is not large and each parameter should be considered individually for the best method especially in light of existing "smart" controller modules. In essence, we believe that CAMAC modules and their brethren are smart enough to take on this chore in a distributed fashion as long as there is an efficient means of synchronizing their activities.

Clock System

The Fermilab clock system runs at 10 MHz and can distribute 256 distinct synchronization events anywhere on the site. There are already a large number of hardware modules which interface to it. A separate clock event is defined to mark the beginning, or reset, of each type of cycle of every accelerator system. For example, MR reset 21 is for 150 GeV injection into the Tevatron, MR reset 29 for antiproton production at 120 GeV, and MR reset 2A a 150 GeV study cycle. By convention, the first digit refers to the accelerator system. Booster events are 1x, where x is any hexadecimal digit from 0 to F, Main Ring 2x, Switchyard 3x, Tevatron 4x, Accumulator-Debuncher are 9x; the Neutron Therapy Facility (NTF),

Preacc, and Linac events are included in the Booster group.

Suppose some hardware parameter register, such as the start of flat-top timer, should change according to which cycle is playing next. Three separate timer registers, each starting on different reset events and loaded with the appropriate delays, would have their outputs summed and connected to whatever hardware needs to know when flat-top occurs. (This particular time is of such wide interest that it is broadcast itself as an event on the accelerator clock system.) These appear as three single devices to the user and each operating mode may be tuned independently. More complex changes which take some time to effect, can be warned in time to make a change by an early signal. All that is left to do is generate the reset events in the desired order. The hardware component which does this is a CAMAC module, the Timeline Generator or TLG.

Timeline Generator Hardware

The TLG is a dual-width CAMAC module based on an in-house single board computer known as a C190-A. It includes a CAMAC interface, common peripheral support chips and an expansion bus for connecting custom modules. In the TLG application a single additional custom circuit card contains expansion RAM memory as well as gating and buffer circuitry for interfacing to the separate clock event encoding hardware.[1]

Firmware in the TLG causes the module to function as a dual 8192-word register where each word is 16 bits wide. One group of 8192 words is used to hold the currently active timeline while the second acts as a communication buffer for receiving a new timeline program from the CAMAC dataway. After the active register has been loaded by console application software, it plays out, one word at a time, every 1/15th of a second. Each word is divided into several coded fields. If a field value is non-zero then that value causes one of several possible accelerator clock resets to be generated and distributed via the clock system. Each field operates independently of the others. The various reset requests (some with a programmable delay) are gated by a 15 Hz signal developed from the power line to maintain synchronism. In the absence of any beam requests, the module generates Booster null cycle resets at 15 Hz. Since most 1/15 second intervals contain no major resets, a compressed data representation is used to extend the maximum length of a timeline program. The 8192 word programs can then supply a timeline which extends in real time for many hours for the types of sequences expected. Figure 1 depicts an overview of the hardware involved.

The module performs several "look-ahead" functions by examining the timeline program ahead of the current instruction. An imminent change in the type of Main Ring cycle is communicated to the Main Ring power supply waveform regulating computers so that appropriate action may be taken. Beam requests of the LINAC system are also examined for several seconds into the future and used to develop a beam permit gate (transmitted on the accelerator clock) to NTF. A permit is generated whenever the LINAC has enough time between its other duties to allow for the rise and fall times of the NTF switching magnet. While these look-ahead functions could certainly be stored in the timeline

program itself, this method capitalizes on easily derived information inherent in the timeline program and economizes on the use of bits in a timeline instruction.
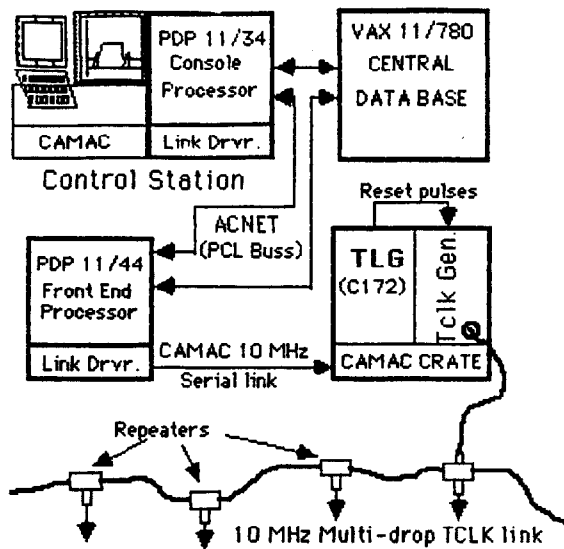

Figure 1 Schematic of the TLG hardware.

CONTROL SOFTWARE

The application program itself, the Timeline Controller or TLC, is designed to allow the machine operator to construct the table of resets and load the table into the module. The general structure of the TLG control program is very much like the program used to control the time and energy dependent parameters of the Tevatron.[2] The TLC is also based on the idea of sequences, or modular calculational algorithms.

Sequence Definitions

Figure 2 shows the console program page which allows the operator to construct or modify the synchronization of the Fermilab accelerator systems. On the right side of the page the detailed clock event structure of a particular sequence is defined. The first appearance of the sequence defines the clock events associated with that sequence even if the sequence is used more than once. (The left side of the page is used to determine the starting time of the sequence and the number of times it repeats. These aspects are covered in the next section.)

The resets necessary for the definition of a sequence are entered with the keyboard. A check is made that the events are logically possible. The number of Booster delay cycles (event 11), prepulses (event 12) and beam events (e.g. 13) are also necessary for the definition of a sequence. The number of sequential Booster pulses is limited to 13 as that is the largest number that the Main Ring can accept.

Take for example the activity of running the Tevatron for Fixed Target physics. The corresponding accelerator clock resets include three Booster pre-pulses (event 12), up to 12 Booster beam pulses (event 13), a MR reset for a 150 GeV ramp (event 21), a TeVatron reset (event 41), and a Switchyard reset (event 31). All of these resets are part of the definition of the TeVFxT sequence shown as the first sequence on figure 2.

Another sequence is for MR studies. This sequence requires Booster prepulses (event 12), one Booster beam (event 13) pulses and a MR reset clock event 29. If one did not want beam from the Booster for the MR studies one could have a MR reset and no Booster resets in the sequence definition.

```
D69  MODIFYing file 2   65 S SC15-8GeV  2D RST  0-150GeV  29 RS
*files  *ch-plot        *TLGTABLE                          *send
*Help                     *CALC/PL  *insert  *delete
     Booster duty factor= 0%        Supercycle period= 90.1333 s
```

SEQ TYPE	Dt	t0	Nt	Fi	Tot	11s	12s	reset clock events #sBR	MR	TV	SY AC
1 TeVFxT	90	0	1	1	90	0	3	8*13	21	41	
2 SYstud	29	18	1	0	47	0	0	0*			31
3 MRstud	7	58	1	0	65	0	3	0*13	2B		
4 Debunc	1	8	18	0	26	0	3	1*15	2D		
5 PbarPR	7	26	3	0	47	8	3	1*13	29		
6 Debunc	1	47	11	0	58	0	3	1*15	2D		

Figure 2. The control page for defining sequences and determining their distribution through the supercycle.

Distribution of Sequences in a Supercycle

Once the sequences have been defined, it remains to determine how they are to occur in time. This is done on the leftmost columns of the console display shown on figure 2. The length of the TLG table is the length of the supercycle; the maximum length is 8192*(1/15) = 546 s = 9.1 minutes. This limit is only a simplification in the application software since it is easier to work with a simple table rather than the compressed table format used by the TLG.

The values needed on the sequence page are:
Dt the sequence duration in seconds. For a Tevatron cycle, for example, this would have to be at least as long as the length of the ramp. Likewise for a sequence involving the MR, the Dt must be at least as long as the length of the corresponding MR ramp.
Tstart the time relative to the start of the super-cycle that you want the sequence to first appear.
Ntimes the number of times you want the sequence to repeat during the supercycle. The duration of the supercycle is determined by the largest Tstart + Ntimes * Dt = ttot of all the sequences. ttot is displayed for each sequence type on the sequence page.
Nfile the number that corresponds to the Tevatron ramp you want to run. At present this number is only used for the plot of reset times.

Calculations

The TLG table is actually calculated and the corresponding plot initialized by an interrupt with the cursor in the left hand column of the sequences. Changing a sequence definition, one of the parameters on the sequence page, or the plot limits zeroes the TLG table. The table is automatically calculated though not plotted if you go to the SEND subpage.

Several error checks are performed on this calculation. The Booster duty factor is ensured to be less than 50% and the calculated value displayed for the operator. There is also a check that no more than one sequence requires a particular Booster reset. Any errors are displayed on the bottom line of the TV screen. One of the more interesting checks is that the supercycle period is a multiple of 15/13 second. This is to ensure synchronization with the central helium liquifier so that the slow spill servo may "learn out" the mechanical motion of some of the nearby ring magnets! If the period does not satisfy the requirement, it is incremented appropriately.

Loading the TLG

The *SEND subpage has controls for loading the TLG as well as for diagnostics and initialization. *Send and Activate New TLG table causes the TLG to be loaded and the new supercycle to start at the end of the present one. If the table is successfully sent, the TLC files are updated with the correct one marked as active.

Plotting

An energy versus time plot is automatically init-ialized when the TLG table is calculated on the sequence page. Figure 3 shows a copy of a Lexidata plot for the sequences defined in Figure 2. Arrows indicate the times of the resets sent by the TLG. The calculated values are plotted in green. The corres-ponding clock events are monitored via the console computer clock interface and are plotted in red with an *. Thus if the sent and read signals coincide, the color is yellow. At the end of a TeVatron cycle the red arrows are erased and the real-time display starts again.

Modify/Save

A typical system of user files is provided to allow saving a timeline for future use or to provide a starting point for construction of new timelines.

OPERATIONAL EXPERIENCE

The number of reset events to be generated by the TLG has grown since first installation. The first version used a simple bit per event mapping in the TLG program words. The module has now come to generate 31 different events by use of encoded bit fields. While it is relatively easy to modify the module firmware and application software to work with wider words, the module has run out of I/O connections. None of this was entirely unexpected. Future versions of the module may cope with this by adding a small external crate for I/O or several TLG modules may operate in parallel.

Varying the intensity of the Booster was easily solved with the addition of timing registers to control the pulse length of the H- beam from the Linac. As anticipated, the dynamic range of the Booster is great enough so that different reset events need only control these registers.

There may be several as yet unknown problems still to be attacked. At least we now have a philos-ophy and a mechanism for making the Fermilab injector a flexible, multipurpose system.

References

[*] Operated by Universities Research Inc., under contract with the United States Department of Energy.

[1] D. Beechy, Fermilab Accelerator Controls Hardware Release 29 (1983), unpublished.

[2] R. W. Goodwin and R. P. Johnson, Proc. of the 12th Int. Conf. On High Energy Accelerators, Fermilab, 1983, p 594.
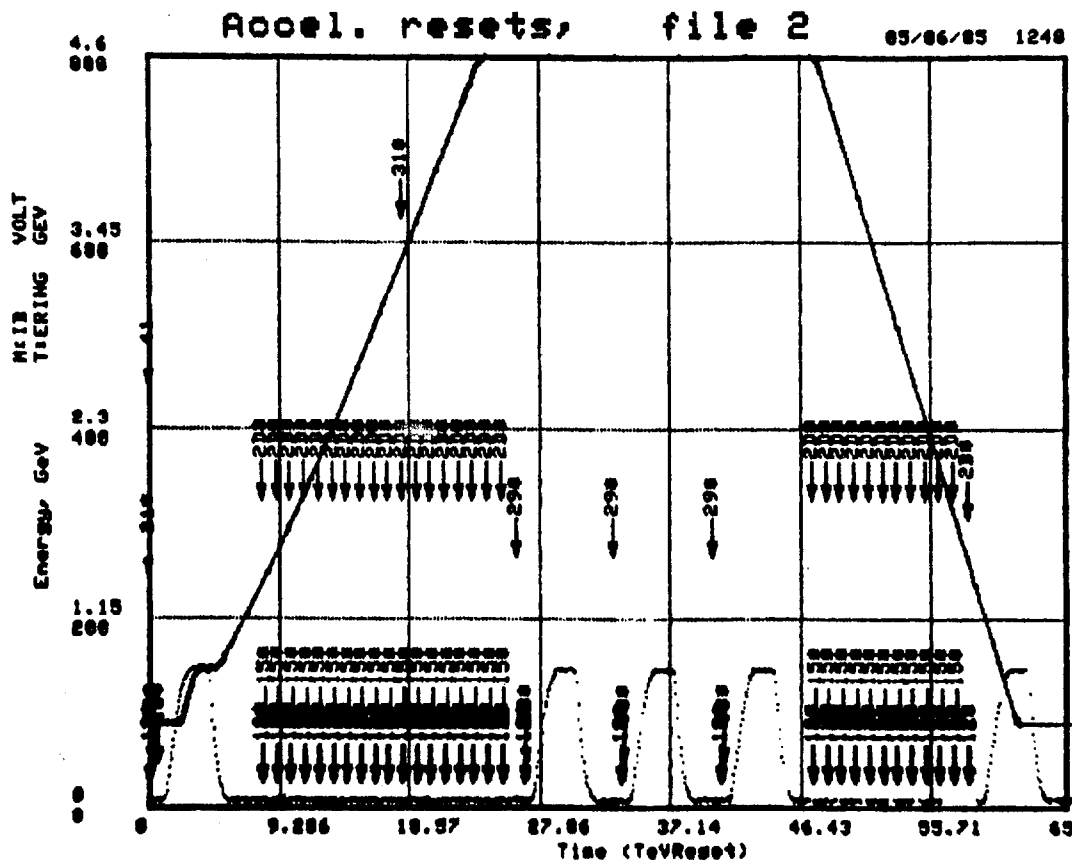
Figure 3. Computed resets and real-time readbacks using time in the supercycle. The ordinate displays the Ring and Tevatron. Clock events are labeled and shown as the console computer clock decoder. The abscissa is the magnetic field and circulating beam current in the Main arrows on the screen.